AMENDMENTS TO THE CLAIMS

1. (Currently amended) A cyphering/decyphering method, by an integrated circuit, of a digital input code by means of several keys, comprising:

dividing said code into several data blocks of same dimensions;

applying to said blocks multiple turns of a cyphering or decyphering comprising submitting each block to at least one same non-linear transformation and of subsequently combining each block with a different key at each turn, and

masking inputs and outputs of the non-linear transformation, upon execution of the method, by means of at least one first random number having the size of said code ~~and all the blocks of which have the same value~~ by combining, by an XOR-type function, the input and output blocks of the non-linear transformation with said at least one first random number,

wherein the at least one first random number comprises a plurality of blocks of bits and wherein each block of bits is identical.

2. (Previously Presented) The method of claim 1, comprising combining the input code with a second random number of same dimension as the code.

3. (Previously Presented) The method of claim 1, wherein said non-linear transformation comprises using a box of substitution of the input code blocks, calculated with a third random number of same length as said code and all the blocks of which have the same value, said box respecting the fact that the transformation of an input code, previously combined by XOR with the first random number, corresponds to the result of the combination by XOR of this input code with said third random number.

4. (Previously Presented) The method of claim 1, applied to an AES-type cyphering algorithm.

5. (Previously Presented) The method of claim 1, wherein said first random number is changed at each cyphering turn.

6. (Previously Presented) The method of claim 2, wherein said second random

number is changed at each cyphering of a new datum.

7. (Previously Presented) The method of claim 3, wherein said third random number is changed at each cyphering turn.

8. (Currently amended) An integrated circuit for cyphering/decyphering by turn input data divided into blocks of same dimensions, comprising:

means for generating at least one first random number~~, comprised of a repeated sequence of a value,~~ of same size as the size of the blocks of the input data; and

means for combining said random number with each block, at an input and at an output of a non-linear transformation implemented by the cyphering/decyphering,

wherein the at least one first random number comprises a plurality of blocks of bits and wherein each block of bits is identical.

9. (Previously Presented) The circuit of claim 8, comprising means for implementing the method of claim 1.

10. (Currently amended) A method comprising:

dividing an input into a plurality of data blocks of the same size;

passing each data block of the plurality through a series of steps, each step applying a non-linear transformation and combining the result of the non-linear transformation with a key specific to the step to generate an output of the step;

combining the output of each step, by an XOR-type function, with a random number having the same size as the output~~ and being comprised of a repeated sequence of a value,~~

wherein the random number comprises a plurality of blocks of bits and wherein each block of bits is identical.

11. (Previously Presented) The method of claim 10, wherein the random number changes at each step.

12. (Previously Presented) The method of claim 10, wherein the random number is a different random number for each step of the series of steps.

13. (Previously Presented) The method of claim 10, wherein the random number is a plurality of random numbers selectively combined with outputs of the series of steps.

14. (Previously Presented) The method of claim 10, applied to an AES-type ciphering algorithm.

15. (New) The method of claim 1, wherein a block of bits is a byte.

16. (New) The circuit of claim 8, wherein a block of bits is a byte.

17. (New) The method of claim 10, wherein a block of bits is a byte.